Neurocomputing 152 (2015) 403–412

Contents lists available at ScienceDirect

Neurocomputing

journal homepage: www.elsevier.com/locate/neucom

Real-time implementation of neural optimal control and state estimation for a linear induction motor

Victor G. Lopez [a], Alma Y. Alanis [b,*], Edgar N. Sanchez [a], Jorge Rivera [b]

[a] CINVESTAV, Unidad Guadalajara, Apartado Postal 31-438, Plaza La Luna, Guadalajara, Jalisco, C.P. 45091, Mexico
[b] CUCEI, Universidad de Guadalajara, Apartado Postal 51-71, Col. Las Aguilas, C.P. 45080, Zapopan, Jalisco, Mexico

ARTICLE INFO

Article history:
Received 23 April 2014
Received in revised form
20 August 2014
Accepted 11 October 2014
Communicated by Xu Zhao
Available online 13 November 2014

Keywords:
Neural optimal control
Recurrent neural networks
Neural observers
Linear induction motor
Real-time implementation
Control Lyapunov function

ABSTRACT

A reduced order state estimator based on recurrent high-order neural networks (RHONN) trained using an extended Kalman filter (EKF) is designed for the magnetic fluxes of a linear induction motor (LIM). The proposed state estimator does not need the mathematical model of the plant. This state estimator is employed to obtain the unmeasurable state variables of the LIM in order to use a state feedback nonlinear controller. A neural inverse optimal control is implemented to achieve trajectory tracking for a position reference. Real-time implementation results on a LIM prototype illustrate the applicability of the proposed scheme.

1. Introduction

Mostly nonlinear state-feedback controllers require system state complete accessibility, which is not always achievable; for this reason, nonlinear state estimation is an important topic. Important applications for nonlinear state estimation are: deterministic and stochastic control [1,2], system modeling [3], fault diagnosis [4,5], and among others. There exists many results for the design of nonlinear state estimators [6,7]; however, these methods do not consider uncertainties [3]. Other studies have offered results on the design of robust nonlinear state estimators [8,9], which still depend on the system model. For real-time applications, a state estimator-controller based on the model of the system may not behave as desired because uncertainties there are always internal and external disturbances, changing parameters and unmodeled dynamics. Neural networks have been established as an appropriate methodology for nonlinear function approximation; then, they can be employed for nonlinear system state estimation [10]. The neural network adapts its synaptic weights in order to adjust its outputs to the system response [11].

These facts have motivated the development of neural-network-based state estimators. In recent years, some of these neural state estimators have been implemented in real-time applications with successful results. In [12,13], neural state estimators are used to obtain the unmeasurable states of the system to be controlled; however, the exact parameters and the mathematical model are required. In [2,14], neural networks are employed to obtain the uncertainties and some parameters of the system model, but not the system states directly; in these cases other parameters of the model must also be known. In [15], a neural state estimator is implemented which consists of a neural identifier for the unknown nonlinear model, and then a conventional Luenberger-like state estimator estimates the system states. In [16] the neural network employed to estimate the system states is trained off-line; this approach has the disadvantage of not being robust against parametric variations.

A different approach for a neural state estimator has been proposed in [11,17]. There, the state estimator is based on a recurrent high-order neural network (RHONN), which is a generalization of the first-order Hopfield network [22]. A RHONN model is easy to implement, has relatively simple structure, is able to adjust its parameters on-line and allows to incorporate a priory information about the system structure [23]. When the neural weights are adapted, the RHONN model dynamics are very similar to the real system dynamics, even in the presence of disturbances. Neither the exact mathematical model, nor the exact parameters, are needed to implement a RHONN.

* Corresponding author.
E-mail address: almayalanis@gmail.com (A.Y. Alanis).

http://dx.doi.org/10.1016/j.neucom.2014.10.031
0925-2312/

Neural networks can be trained by many different algorithms [24]; however, most of these algorithms normally encounter technical problems such as local minima, slow learning and high sensitivity to initial conditions, among others [11]. The Extended Kalman Filter (EKF) form the basis of a second-order neural network training method [25], where the synaptic weights become the states to be estimated. The EKF training algorithm provides a recursive optimal estimator for the neural weights.

Optimal control of nonlinear systems deals with obtaining a control law for a given system such that a cost functional is minimized [26], Dynamic programming, developed by Bellman [27], is a solution for optimal control which leads to a nonlinear partial differential equation named as the *Hamilton–Jacobi–Bellman* (HJB) equation. Solving this equation is not straightforward: for systems of dimension higher than two there are no practical ways to solve such equation [28]. Inverse optimal control is an alternative for optimal control synthesis which avoids the need to solve the associated HJB equation [28,29]. For the inverse approach, a stabilizing feedback control law, based on a priory knowledge of a Control Lyapunov Function (CLF), is designed first and then it is established that this control law optimizes a cost functional. In this paper, we propose to integrate a reduced order RHONN state estimator and an inverse optimal control law based on a CLF for nonlinear systems.

This mentioned control scheme is applied to a linear induction motor (LIM). The LIM is a linear electric actuator on which the electrical energy is turned into mechanical translational movement; this is, a mobile element on the motor moves linearly with respect to a stationary element [30]. Induction motors have been widely studied and several control approaches have been applied to them [31–33]. LIMs present advantages with respect to other types of motors. They develop magnetic forces directly between the mobile element and the stationary element, without the need of physical contact between both elements, which would restrict the system dynamics. Then, LIM can reach higher speed and reduces undesirables vibrations [30]. For these reasons the LIM has been employed widely in industrial applications such as steel, textile, nuclear and space industries [34]. However, the most extensive application for LIMs is for public transportation by means of high speed trains. The idea of using linear motors for mass public transportation is not new. However, the attention is focused on the LIM again due to recent developments of smart grids. These are complex systems with unknown uncertainties and disturbances [35], which cause that the control synthesis can be very difficult to handle with traditional approaches, requiring the application of intelligent control ones [17].

Control of Rotational Induction Motor (RIM) has been extensively studied, which is not the case for LIM, even if driving principles of both kind of motors are similar. However, recently different control techniques have been developed for LIM. For instance in [18] an adaptive backstepping-sliding mode controller is proposed, in [19] a fuzzy sliding-mode controller is implemented in a field-programmable gate array, and then in [20] a robust controller is proposed to relax disturbances requirements with the fusion of an integral-proportional position control and neural network to estimate disturbances. In [21] is established a field-oriented control scheme, considering the end effect. For those controllers the design is developed for continuous-time and implemented experimentally for position trajectory tracking. Although, those controllers are robust to uncertainties, they require previous knowledge of plant model and/or plant parameters at least their nominal values. Besides, the progresses in digital equipment have attracted considerable efforts to the design of high performance discrete-time controllers for continuous-time plants, which has not been studied deeply as the continuous-time ones [18–21].

The main contributions of the paper are: first a novel inverse optimal neurocontroller with control gain matrix reduction for systems that are or can be written in the nonlinear block controllable form; second, the stability proof on Lyapunov basis for the proposed controller, then the use of a reduced order neural observer to relax the full state measurement assumption. Finally the real-time implementation of the proposed inverse optimal control scheme and the neural state estimator on a LIM prototype in order to show the effectiveness of the proposed scheme, for a nonlinear system under internal and external disturbances which model, parameters and uncertainties are considered unknown and with partial state measurements.

In the following, Section 2 presents mathematical preliminaries for the neural networks state estimation. Section 3 includes the inverse optimal control basis. In Section 4 the RHONN identifier is explained. Sections 5 describes the neural inverse optimal control application for LIM. Section 6 presents the real-time implementation results and Section 7 exposes the respective conclusions.

2. Neural networks state estimation

Through this paper, subindex k is used as the sampling time, with $k \in \{0\} \cup \mathbb{Z}^+$.

In this paper, we consider the discrete-time multiple-input multiple-output nonlinear system

$$\chi_{k+1} = F(\chi_k, u_k) \tag{1}$$

where $\chi_k \in \Re^n$ is the state of the system, $u_k \in \Re^m$ is the control input and $f(\chi_k) : \Re^n \to \Re^n$ is a smooth map.

2.1. Recurrent high-order neural networks

For practical situations, the mathematical model of the system to be controlled is usually unknown; then, a RHONN identifier is employed to obtain a neural model of the system, required for the implementation of the control law. The discrete-time RHONN employed for identification of a nonlinear system (1) is defined as

$$x_{i,k+1} = w_i \phi_i(x_k, u_k), \quad i = 1, \dots, n \tag{2}$$

where x_i is the state of the ith neuron, w_i is the respective online adapted weight vector, n is the system state dimension and $\phi_i(x_k, u_k)$ is given by

$$\phi_i(x_k, u_k) = \begin{bmatrix} \phi_{i_1} \\ \phi_{i_2} \\ \vdots \\ \phi_{i_{L_i}} \end{bmatrix} = \begin{bmatrix} \prod_{j \in I_1} y_{i_j}^{d_{i_j}(1)} \\ \prod_{j \in I_2} y_{i_j}^{d_{i_j}(2)} \\ \vdots \\ \prod_{j \in I_{L_i}} y_{i_j}^{d_{i_j}(L_i)} \end{bmatrix} \tag{3}$$

with L_i being the respective number of high-order connections, $I_1, I_2, \dots, I_{L_i}$ is a collection of non-ordered subsets of dimension $1, 2, \dots, n+m$, m is the number of external inputs, d_{i_j} is a non-negative integer and y_i is defined as follows:

$$y_i = \begin{bmatrix} y_{i_1} \\ \vdots \\ y_{i_n} \\ y_{i_{n+1}} \\ \vdots \\ y_{i_{n+m}} \end{bmatrix} = \begin{bmatrix} S(x_1) \\ \vdots \\ S(x_n) \\ u_1 \\ \vdots \\ u_m \end{bmatrix} \tag{4}$$

In (4), $u_k = [u_1, \dots, u_m]^T$ is the input vector to the RHONN and $S(\cdot)$ is defined by

$$S(x) = \tanh(\gamma x) \tag{5}$$